Boston_Map

Longhao

November 3, 2018

The first step is to import the data file downloaded from the website.

```
mayorsfoodcourt <- read.csv("mayorsfoodcourt.csv")
```

Then we are interested in how many food courts available in Boston and we will need to do some data cleaning.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.0.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.4
## v tidyr   0.8.1     v stringr 1.3.0
## v readr   1.1.1     v forcats 0.3.0
```

```
## -- Conflicts ------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(maps)
```

```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

```
library(mapdata)
library(ggmap)
# This is find out the unique food courts in the dataframe.
total <-unique(x = mayorsfoodcourt$LICENSENO)

#There are total 8168 food courts in Boston Areas. But not all of them are active
#so we want to remove the inactive courts. Also not all of them have a record on violation status
#we want to remove those rows as well.There are data from 2007 until 2018,
#we are probbaly only interested in the recent 2 years data so we want to filter the data.
#
act_Court<- mayorsfoodcourt %>% filter(LICSTATUS == "Active") %>% filter(ViolStatus == "Pass"| ViolStatu

#We want to convert location into latitude and longitude and remove NA values since we can not plot the
#After calculation there is about 28% of locations missing from dataset.
locat1<-act_Court%>% separate(Location, c("latitude", "longitude"), ",") %>% filter(!is.na(longitude))
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 22874 rows
## [70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87,
## 88, 89, ...].
```

```
#There are annoying charact or "(", ")" in the column entry and we want to remove them
locat1$latitude<-as.numeric(substr(locat1$latitude,start = 2,stop = 13))
locat1$longitude<-as.numeric(substr(locat1$longitude,start = 1, stop = 13))
```

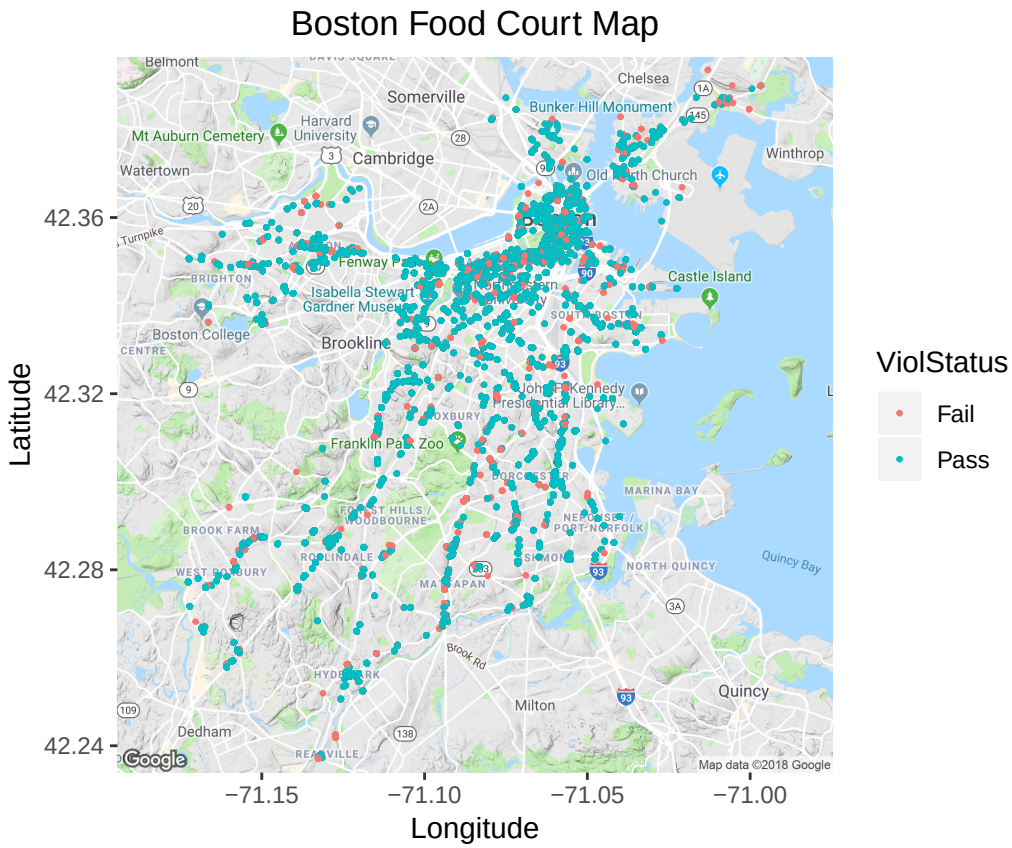
```r
#Then we want to look at individual business pass/fail status
locat1$pass_fail <- as.numeric(locat1$ViolStatus)-3
locat1$score<-rep(NA,dim(locat1)[1])
locat1$score[which(locat1$pass_fail==0)]<--1
locat1$score[which(locat1$pass_fail==1)]<-1
locat1$ViolLevel<-as.numeric(locat1$ViolLevel)-2
same_food_court <- locat1 %>% group_by(LICENSENO)
sum<-same_food_court %>%
  summarise(passrate=mean(pass_fail))
sum1<-same_food_court %>%
  summarise(total_score=sum(ViolLevel*score))

long<-locat1 %>%
  select("latitude","longitude","LICENSENO") %>%
  unique()

tscore_data<-data.frame(left_join(sum1,long,by="LICENSENO"))
rate_data<-sum %>% left_join(long,by="LICENSENO")

#Then let's define the boundry
boston_bbox <- make_bbox(lat = latitude, lon = longitude, data = locat1)



#If you run into error for the following code, it is not because the code is wrong.
#This is due to the fact that Google changes their API policy and we can no longer
#get access through the Goolge Api without billing them and there is also a daily limit.

boston_map <- get_map(location = boston_bbox, source = "google", maptype = "terrain")
```

```
## Warning: bounding box given to google - spatial extent only approximate.

## converting bounding box to center/zoom specification. (experimental)

## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=42.315331,-71.08467&zoom=12&size=
```

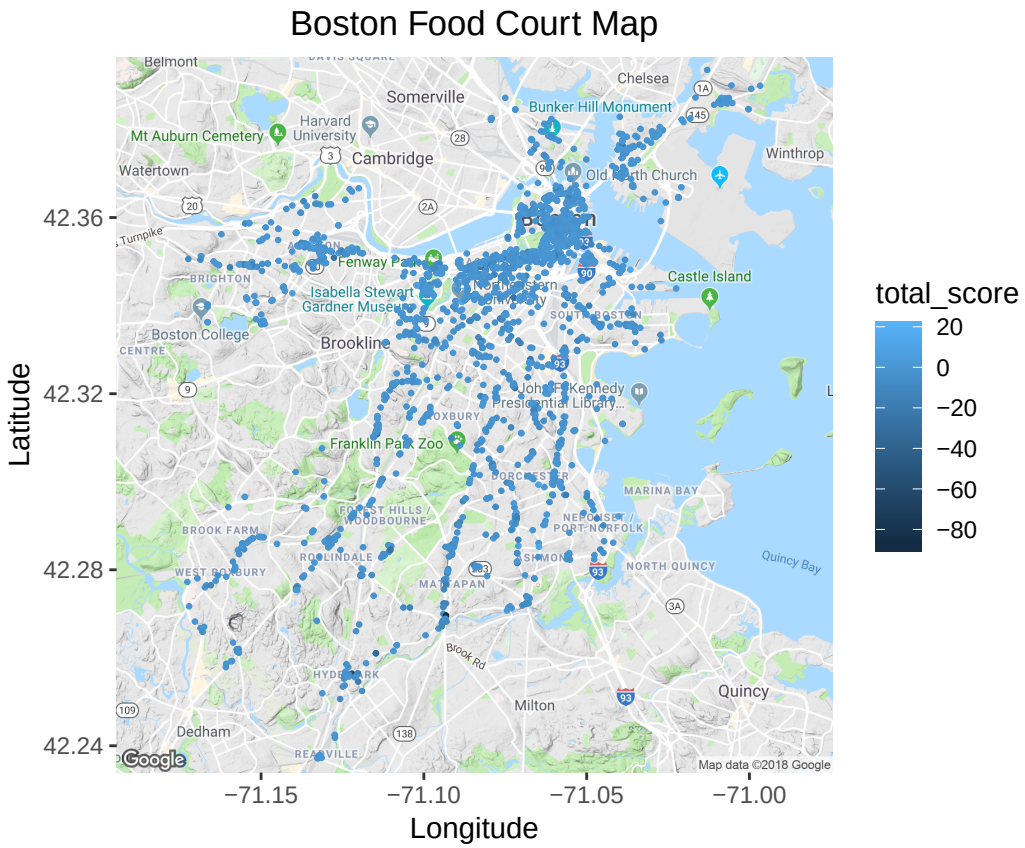
```r
bc_big <- get_map(location = boston_bbox, source = "google", maptype = "terrain")
```

```
## Warning: bounding box given to google - spatial extent only approximate.

## converting bounding box to center/zoom specification. (experimental)
## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=42.315331,-71.08467&zoom=12&size=
```

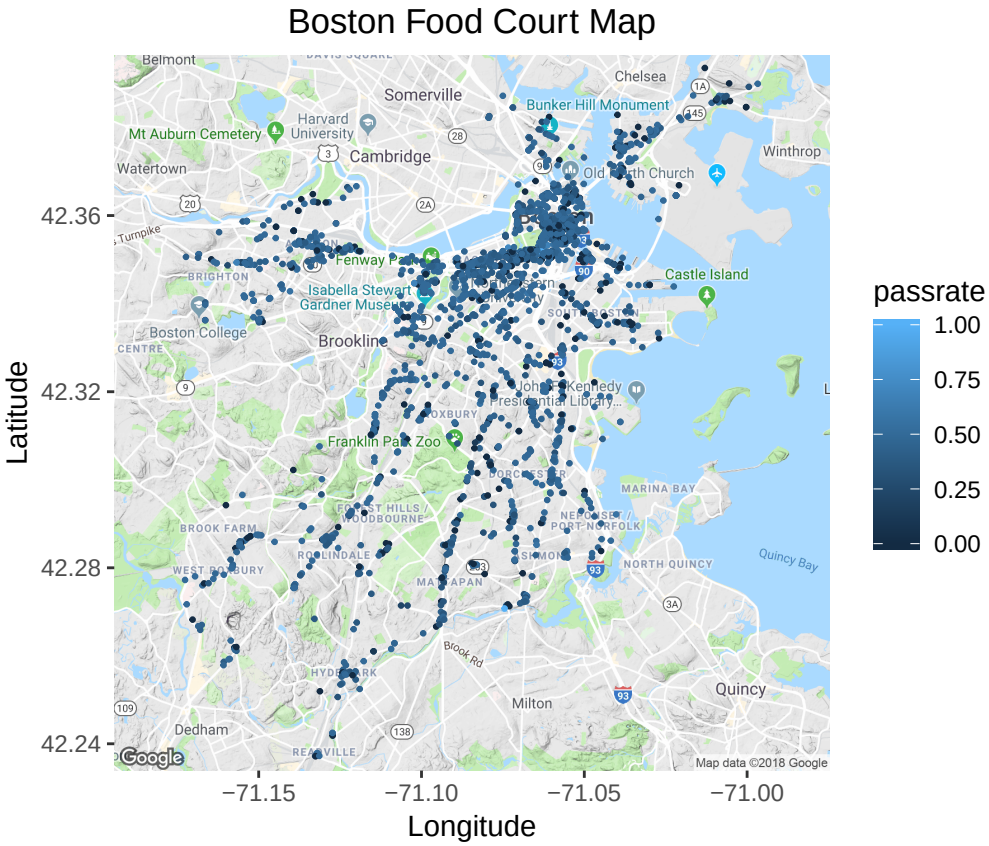
```r
ggmap(boston_map) +
  geom_point(data = locat1, mapping = aes(x = longitude, y = latitude, color= ViolStatus),size=0.5)+
  xlab("Longitude")+ylab("Latitude") +
  ggtitle("Boston Food Court Map") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
ggmap(boston_map) +
  geom_point(data = tscore_data, mapping = aes(x = longitude, y = latitude, color= total_score),size=0.5
  xlab("Longitude")+ylab("Latitude") +
  ggtitle("Boston Food Court Map") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
ggmap(boston_map) +
  geom_point(data = rate_data, mapping = aes(x = longitude, y = latitude, color= passrate),size=0.5)+
  xlab("Longitude")+ylab("Latitude") +
  ggtitle("Boston Food Court Map") +
  theme(plot.title = element_text(hjust = 0.5))
```


Boston Food Court Map
Latitude
42.36
42.32
42.28
42.24
−71.15
−71.10
−71.05
−71.00
Longitude
passrate
1.00
0.75
0.50
0.25
0.00
Belmont
Somerville
Chelsea
Bunker Hill Monument
Mt Auburn Cemetery
Harvard University
Cambridge
Watertown
Winthrop
Old North Church
Fenway Park
Castle Island
Isabella Stewart Gardner Museum
Boston College
Brookline
Franklin Park Zoo
MARINA BAY
NORTH QUINCY
Quincy Bay
Brook Rd
Milton
Quincy
Dedham
Google
Map data ©2018 Google